Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in this Application:

Listing of Claims:

1. (Currently amended) A distributed object-oriented software development environment comprising:

an application network comprising a plurality of objects for performing object operations, each object including an object interface;

at least one object port coupled to said each object interface of said objects; and interaction means for connecting said object port of one of said objects to said object port of another one of said objects,

wherein one of said objects can communicate to another one of said objects if said object interfaces are compatible and said interaction means provides sequential flow of data and control from said object operations through a dynamically varying set of said ports having said ~~compatible~~ object interfaces which are compatible for performing consistent and transport dynamic updates of said application network.

2. (Original) The environment of claim 1 wherein said interaction means is represented by a circular communication pathway and a first said object port is connected to said circular communication pathway to receive communications from at least a second said object port which is connected to said circular communication pathway.

3. (Original) The environment of claim 1 wherein said interface is described in modified CORBA interface description language.

4. (Currently amended) The environment of claim 1 further comprising:

a plurality of management objects, each said management object being associated with at least one of said objects;

a human manager object; and

an interface for network evolution for coupling said management objects to said human manager object, wherein said human manager object manages said objects for performing object operations through said management objects.

5. (Currently amended) The environment of claim 4 wherein said human manager object assigns increasing object version numbers to said objects for performing object operations.

6. (Original) The environment of claim 5 wherein said human manager object assigns ~~monotomically~~monotonically increasing interface versions to said object interfaces wherein each said object interface has a unique global identification in said application network.

7. (Currently amended) The environment of claim 6 further comprising:

means for determining said ~~compatible~~ ports having said object interfaces which are compatible interfaces of said objects for performing object operations by registering said global identification and said object version number of said object for performing object operations with said management object.

8. (Currently amended) The environment of claim 7 further comprising:

means for determining an object table comprising rows representing said object versions of said objects for performing object operations in said network application and columns representing an object identification and interface identification;

means for sorting said determined object table with respect to an object version;

means for sorting a first said sorted object table for a first said object for performing object operations and a second said sorted object table for a second said object with respect to a common said interface identification;

means for joining said first said sorted object table and said second said sorted object for performing object operations with respect to said interface identification; and

means for extracting said compatible object from said join of said object tables.

9. (Original) The environment of claim 8 further comprising:

means for sorting a subsequent object table with respect to said common said interface identification; and

means for joining said subsequent object table with said joined first said sorted object table and said second said sorted object table.

10. (Currently amended) The environment of claim 1 further comprising a life cycle framework including a specification stage in which said objects for performing object operations

and said interfaces are specified, a design stage in which said interfaces of said objects for performing object operations are negotiated, an implementation stage in which said negotiated interfaces of said objects for performing object operations are implemented and a testing stage in which said implemented interfaces are tested.

11. (Currently amended) A method for implementing negotiation during software development comprising the steps of:

a. determining a human manager object;

b. determining at least one management object;

c. determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object, by instructing first objects with said at least one management object to create a plurality of first objects for performing object operations, each said first object including an object interface,

d. creating an interaction means for connecting said at least one first object to said management objects;

e. determining at least one management object port associated with said management object;

f. determining at least one object port associated with said first object; and

g. forwarding ~~negotiations~~ negotiation scripts from said object ports to said management object ports.

12. (Currently amended) The method of claim 11 further comprising the step of:

assigning tasks of designing said first objects from said human manager object to a respective developers associated with at least one of said first objects.

13. (Currently amended) The method of claim 12 further comprising the step of:

creating a developer negotiation port by said developer for each of said first objects to be developed.

14. (Original) The method of 13 further comprising the step of:

registering said developer negotiation ports with said human manager object.

15. (Original) The method of claim 14 further comprising:

creating management negotiation ports at said management objects which are each associated respectively with one of said developer negotiation ports.

16. (Currently amended) The method of claim 15 wherein step g comprises:

forwarding negotiation scripts written in modified CORBA IDL by said developers through said respective developer negotiation ports to said respective manager negotiation ports for forwarding to designated said first objects.

17. (Original) The method of claim 16 further comprising the step of:

forwarding said scripts written in modified CORBA IDL received at said management object to said human manager object via said INE.

18. (Original) The method of claim 17 further comprising the step of:

interpreting said script written in modified CORBA IDL received at said human manager object into human readable data.

19. (Original) The method of claim 11 wherein the step of forwarding negotiations is repeated until all developers have agreed.

20. (Original) The method of claim 19 wherein said negotiations determine an object interface defined in modified CORBA IDL.

21. (Currently amended) A method for implementing a network application comprising the steps of:

determining a plurality of first objects;

associating an object port with each of said first objects;

determining transactions for exchanging messages between said first objects;

determining an object interface for each said first object; and

implementing each determined object interface, wherein said messages are exchanged sequentially between said first objects having compatible said object interfaces.

22. (Original) The method of claim 21 further comprising the step of:

registering said implemented object and said object interface with a management framework, said management framework returning an object identification and an object version identification and an interface version identification.

23. (Original) The method of claim 22 wherein said implementing step further comprises the step of:

determining a network application having compatible said object version identifications.

24. (Original) The method of claim 23 wherein said step of:

determining a network application having compatible object versions comprises the steps of:

a. determining an object table comprising rows representing said object identification and said object version identification and columns representing said interface version identification;

b. sorting said determined object table with respect to said object version identification;

c. sorting a first said sorted object table for a first said object and a second said sorted object table for a second said object with respect to a common said interface identification;

d. joining said first said sorted object table and said second said sorted object with respect to said interface identification to form a join of said object tables; and

e. extracting said compatible object from said join of said object tables.

25. (Original) The method of claim 24 further comprising the steps of:

f. sorting a subsequent object table with respect to said common said interface identification; and

g. joining said subsequent object table with said joined object table of step (d);.

26. (Original) The method of claim 24 wherein said object tables are created to have said object version identification and said interface version identification increasing in said rows and said columns.

27. (Currently amended) The method of claim 21 further comprising the steps of:

determining a plurality of management objects, each said management object being associated with at least one of said first objects;

determining a human manager object;

determining an interface for network evolution for coupling said management objects to said human manager object; and

managing said first objects by said human manager object through interacting with said management objects.

28. (Currently amended) The method of claim 27 further comprising the steps of:

updating said determined first objects; and

assigning increasing object version numbers by said human manager object to said updated first objects through said management objects.

29. (Currently amended) The method of claim 27 further comprising the step of:

updating said object interface; and

assigning increasing interface version numbers by said human manager object to said an updated said first object through said management objects.

30. (Currently amended) A method for setting up a network application comprising the steps of:

a. determining a human manager object;

b. determining at least one management object;

c. determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object; instructing said at least one management object by said human manager object to create at least one first object for performing object operations, each said first object for performing object operations including an object interface,

d. creating an interaction means for connecting said first objects to said management objects, said interaction means also being connected to said INE and said human manager object; and

e. initializing states at said human manager object of said first objects and forwarding said initialized states to said first objects via said INE to forward to said initialized states to said management object and said management object forwarding said initialized states from said management object to said first objects.

31. (Original) The method of claim 30 after step c further comprising the steps of:

f. determining a human manager object INE port for said human manager object;

g. determining a management object INE port for said management object; and

h. associating said INE with said INE port for said management object and said INE port for said manager object.

32. (Currently amended) The method of claim 31 further comprising the steps of:

determining at least one port associated with said management object; and

determining at least one object port associated with each said first object.

33. (Original) The method of claim 30 wherein said object interface is defined in modified CORBA IDL.

34. (Currently amended) A method for dynamically reconfiguring a network application comprising the steps of:

determining a human manager object;

determining at least one management object;

determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object, by instructing objects with said at least one management object to create at least one first object for performing object operations, each said first object including an object interface and having an original state,

creating an interaction means for connecting said at least one first object to said management objects;

determining at least one management object port associated with said management object;

determining at least one object port associated with said first object; and

establishing ~~quiesent~~quiescent points in at least one of said first objects to be reconfigured through said management object.

35. (Currently amended) The method of claim 34 further comprising the step of:

forwarding data for updating said at least one object from said first object to said human manager object.

36. (Currently amended) The method of claim 35 further comprising the steps of:

determining said port of said first object to be reconfigured;

sending a destroy command from said human manager object to destroy said port to be reconfigured;

creating a new version of said first object to be reconfigured at said human manager object;

forwarding said new version of said first object to said management object;

creating a new object having said new version of said first object; and

determining a new object port associated with said new object.

37. (Currently amended) The method of claim 36 further comprising the steps of:

determining at said human manager object if a said original state of said first object is the same as a state of said new version of said object; and

if said original object version and said new version have the same states, replacing said original object version with said new version; or

if said original object version and said new version do not have the same state, determining at said human manager object an equivalent state and replacing said original version with said new version.

38. (Currently amended) The method of claim 37 further comprising the step of:

forwarding data for updating said at least one interface version from one of said first objects to said human manager object.

39. (Currently amended) The method of claim 38 further comprising the steps of:

determining a number of said first objects to be reconfigured for said updating of said interface version;

sending a destroy command from said human manager to destroy said number of first objects to be reconfigured;

creating a new version of each said number of first objects to be reconfigured at said human manager object;

forwarding said new versions to said management object; and

creating a corresponding number of new objects having said new versions.

40. (Original) The method of claim 34 wherein said object interface is defined in modified CORBA IDL.

41. (Currently amended) A system for implementing negotiation during software development comprising:

an application network;

means for determining a human manager object;

means for determining at least one management object;

means for determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object, by instructing objects with said at least one management object to create a plurality of first objects for performing object operations, each said first object including an object interface;

means for creating an interaction means for connecting said at least one first object to said management objects;

means for determining at least one management object port associated with said management object;

means for determining at least one object port associated with said first object; and

means for forwarding negotiations from said object ports to said management object ports which are compatible for performing consistent and dynamic updates of said application network.

42. (Currently amended) The system of claim 41 further comprising:

means for creating a developer negotiation port by said developer for each of said first objects to be developed.

43. (Original) The system of claim 42 further comprising:

means for registering said developer negotiation ports with said human manager object.

44. (Original) The system of claim 43 further comprising:

means for creating management negotiation ports at said management objects which are each associated respectively with one of said developer negotiation ports.

45. (Original) The system of claim 44 wherein said negotiations are written in modified CORBA IDL.

46. (Original) A system for implementing a network application comprising:

means for determining a plurality of objects; means for associating an object port with each of said objects;

means for determining transactions for exchanging messages between said objects;

means for determining an object interface for each said object; and

means for implementing each determined object interface, wherein said messages are exchanged sequentially between said objects having compatible said object interfaces.

47. (Original) The system of claim 46 further comprising:

means for registering said implemented object and said object interface with a management framework, said management framework returning an object identification and an object version identification and an interface version identification.

48. (Original) The system of claim 47 wherein said means for implementing comprises:

means for determining a network application having compatible said object version identifications.

49. (Original) The system of claim 48 wherein said means for determining a network application having compatible object versions comprises:

means for determining an object table comprising rows representing said object identification and said object version identification and columns representing said interface version identification;

means for sorting said determined object table with respect to said object version identification;

means for sorting a first said sorted object table for a first said object and a second said sorted object table for a second said object with respect to a common said interface identification;

means for joining said first said sorted object table and said second said sorted object with respect to said interface identification to form a join of said object tables; and

means for extracting said compatible object from said join of said object tables.

50. (Original) The system of claim 46 wherein said object interface is defined in modified CORBA IDL.

51. (Currently amended) A system for setting up a network application comprising:

means for determining a human manager object;

means for determining at least one management object;

means for determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object;

instructing said at least one management object by said human manager object to create at least one first object for performing object operations, each said at least one first object including an object interface,

means for creating an interaction means for connecting said first objects to said management objects, said interaction means also being connected to said INE and said human manager object; and

means for initializing states at said human manager object of said first objects and forwarding said initialized states to said first objects via said INE to forward to said initialized states to said management object and said management object forwarding said initialized states from said management object to said first objects.

52. (Original) The system of claim 51 further comprising:

means for determining a human manager object INE port for said human manager object;

means for determining a management object INE port for said management object; and

means for associating said INE with said INE port for said management object and said INE port for said manager object.

53. (Currently amended) The system of claim 52 further comprising:

means for determining at least one port associated with said management object; and

means for determining at least one object port associated with each said first object.

54. (Original) The system of claim 51 wherein said object interface is defined in modified CORBA IDL.

55. (Original) A system for dynamically reconfiguring a network application comprising:

means for determining a human manager object;

means for determining at least one management object;

means for determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object, by instructing objects with said at least one management object to create at least one first object for performing object operations, each said first object including an object interface and having an original state,

means for creating an interaction means for connecting said at least one first object to said management objects;

means for determining at least one management object port associated with said management object;

means for determining at least one object port associated with said at least one first object; and

means for establishing ~~quiesent~~quiescent points in at least one of said at least one first objects to be reconfigured through said management object.

56. (Currently amended) The system of claim 55 further comprising:

means for forwarding data for updating said at least one first object from said first object to said human manager object~~.~~;

means for determining said port of said first object to be reconfigured;

means for sending a destroy command from said human manager object to destroy said port to be reconfigured;

means for creating a new version of said first object to be reconfigured at said human manager object;

means for forwarding said new version of said first object to said management object;

means for creating a new object having said new version of said first object; and

means for determining a new object port associated with said new object.

57. (Currently amended) The system of claim 56 further comprising:

means for forwarding data for updating said at least one interface version from one of said first objects to said human manager;

means for determining a number of said first objects to be reconfigured for said updating of said interface version;

means for sending a destroy command from said human manager to destroy said number of first objects to be reconfigured;

means for creating a new version of each said number of first objects to be reconfigured at said human manager object;

means for forwarding said new versions to said management object;

means for creating a corresponding number of new objects having said new versions.

58. (Original) The system of claim 55 wherein said object interface is defined in modified CORBA IDL.